COMPUTING L-FUNCTIONS
WITH LARGE CONDUCTOR

JEFFREY STOPPLE

ABSTRACT. An algorithm is given to efficiently compute L-functions with large conductor in a restricted range of the critical strip. Examples are included for about 21000 dihedral Galois representations with conductor near 10^7. The data shows good agreement with a symplectic random matrix model.

1. INTRODUCTION.

In [12], Odlyzko and Schönhage developed an algorithm to compute the Riemann zeta function $\zeta(s)$ efficiently for values of s very high up in the critical strip. Their method depends on precomputation of Taylor series expansions of $\zeta(s)$ at regularly spaced points, which in turn can be done efficiently by a clever application of the Fast Fourier Transform. Rumely later implemented a version of this for Dirichlet L-functions in [16].

In analytic number theory it is often the case that there is a symmetry between what one can prove for large values of $t = \text{Im}(s)$ in the critical strip, and what one can prove for L-functions with large conductor q. With this in mind, it seems reasonable to try to find an algorithm to efficiently compute values of L-functions with very large conductor.

Roughly speaking, we want to view the L-function as a Mellin transform of an automorphic form f, and split the integral at the symmetry point. This gives the extended L-function as an infinite sum

$$\Lambda(s, f) = \sum_n a(n)\{G(s, 2\pi n/q^{1/2}) + G(1 - s, 2\pi n/q^{1/2})\}$$

of values of an incomplete Gamma function

$$G(s, x) = x^{-s}\Gamma(s, x) = \int_1^\infty \exp(-yx)y^s \frac{dy}{y}.$$

It was already observed in [17] that the rapid decay of $G(s, x)$ as $x \to \infty$ means that we can truncate the infinite series at about $O(q^{1/2})$ terms. On the other hand, the exponential decay as $t = \text{Im}(s)$ increases will cause a loss of precision if we want to compute L-function values for large t. This can be fixed by moving the contour integral so that the x parameter has complex values, as observed in [10] and [8] and implemented in [15].

The algorithm is based on computing Taylor series expansions of the function $G(s, x)$ in the *second* variable. Differentiation under the integral sign and integration by parts gives a recursion which allows the computations of derivatives of $G(s, x)$ at very little cost. The rapid decay of $G(s, x)$ as x increases will imply that, instead of taking equally spaced points for the centers of the Taylor expansions, the sequence of points can grow exponentially. This, in turn, will imply that we need very few Taylor expansions to compute efficiently, in fact $O(\log(q))$ expansions with $O(\log(q))$ terms in each.

However, too much change in t will require that we increase the phase in the x parameters, and then all Taylor expansions must be recomputed. For that reason, we will only consider $|t| \leq 1$ in this paper, although the method will generalize to any bounded region in the critical strip. For further simplification we will consider only $s = 1/2 + it$ on the critical line, although this, too, is not crucial.

If we want to get the most benefit of the precomputations it makes sense to compute zeros of lots of L-functions with the same conductor simultaneously. Here the power of the Fast Fourier Transform can again play a role, if all of the L-functions arise from the same abelian group structure. In this paper we will focus primarily on L-functions attached to characters ϕ on the ideal class group of a complex quadratic field with discriminant $-q$. The corresponding automorphic forms are weight one forms, which are cusp forms if the character ϕ is not a genus character. The corresponding Galois representations are dihedral. (The final section of the paper, however, includes an example of an elliptic curve L-function with large conductor, and of an L-function for a quadratic Dirichlet character for large discriminant.)

With this in mind we now explain the algorithm in a little more detail. It will be convenient to use the correspondence between ideal classes and binary quadratic forms Q of discriminant $-q$. Corresponding to a character ϕ we have a theta function

$$\Theta(z, \phi) = \sum_{[Q]} \phi(Q) \sum_{n>0} r_Q(n) \exp(2\pi i n z) \stackrel{\text{def.}}{=} \sum_{n>0} r_\phi(n) \exp(2\pi i n z).$$

Then

$$\Lambda(s, \phi) \stackrel{\text{def.}}{=} (q^{1/2}/2\pi)^s \Gamma(s) L(s, \phi) =$$
$$\sum_{n>0} r_\phi(n) \left\{ G(s, 2\pi n/q^{1/2}) + G(1-s, 2\pi n/q^{1/2}) \right\}.$$

(See [5, §12.4] or [6, §22.3].) For purposes of locating zeros on the critical line it is better to compute the Hardy function $Z(t, \phi)$ defined for $t > 0$ by

$$\theta(t, \phi) = t \log(q^{1/2}/(2\pi)) + \arg(\Gamma(1/2 + it))$$

and, with $s = 1/2 + it$,

$$Z(t, \phi) = \exp(i\theta(t, \phi)) L(s, \phi)$$
$$= \exp(i\theta(t, \phi))(2\pi/q^{1/2})^s \Gamma(s)^{-1} \times$$
$$\sum_{n>0} r_\phi(n) \left\{ G(s, 2\pi n/q^{1/2}) + G(1-s, 2\pi n/q^{1/2}) \right\}.$$

We will truncate the infinite series after N terms, and arrange the $n < N$ into intervals $I_j, j = 1, 2, \ldots T$ centered at points x_j and with width Δ_j. For n in the interval I_j we compute a Taylor series expansion, truncated to B terms

$$G(s, 2\pi n/q^{1/2}) + G(1-s, 2\pi n/q^{1/2}) \approx \sum_{k=0}^{B} G_{j,k}(s)(2\pi n/q^{1/2} - x_j)^k,$$

where

(1) $$G_{j,k}(s) = \left\{ G^{(k)}(s, x_j) + G^{(k)}(1-s, x_j) \right\} / k!.$$

This gives $Z(t, \phi)$ as

(2) $$Z(t, \phi) \approx \exp(i\theta(t, \phi))(2\pi/q^{1/2})^s \Gamma(s)^{-1} \times$$
$$\sum_{j=1}^{T} \sum_{k=0}^{B} G_{j,k}(s) \Delta_j^k \sum_{n \in I_j} r_\phi(n) \frac{(2\pi n/q^{1/2} - x_j)^k}{\Delta_j^k}.$$

Here we have inserted canceling terms Δ_j^k, Δ_j^{-k} to control the small size of $G_{j,k}(s)$ and the large size of $(2\pi n/q^{1/2} - x_j)^k$.

The point, then, is that the inner sum over $n \in I_j$ is independent of s, and so can be done as a precomputation. Of course, this is done separately for each character ϕ, but the coefficients $r_\phi(n)$ can be computed for all characters ϕ very efficiently via the Fast Fourier Transform once the representation numbers $r_Q(n)$ are known.

If we want to use this method for other kinds of L-functions, for example, the L-function of an elliptic curve, or a quadratic Dirichlet

character, we forgo the advantage of using the Fast Fourier Transform. However, we see below we can choose the parameters x_j to be an integer times (a constant analogous to) $2\pi/q^{1/2}$, so that the pre-computation can even be done with *integer* arithmetic when the coefficients $r(n)$ are integers. The Δ_j^k terms are simply omitted, and the (analog of the) $(2\pi/q^{1/2})^k$ term is factored out of the inner sum. The author would like to thank Kimberly Hopkins for help with this idea.

The remaining sections of the paper are as follows:

2. Review of properties of $G(s, x)$.
3. Truncation of the L-series after N terms.
4. Arrangement of the Taylor expansions.
   The T intervals $I_j = (x_j - \Delta_j/2, x_j + \Delta_j/2)$ are determined.
5. Truncation of the Taylor expansions after B terms.
6. Implementation and examples.

2. REVIEW OF PROPERTIES OF $G(s, x)$.

As above we define for $x > 0$ and $s \in \mathbb{C}$

$$G(s, x) = x^{-s}\Gamma(s, x) = \int_1^\infty \exp(-xy)y^s \frac{dy}{y}.$$

Differentiating with respect to x under the integral we see that

$$(3) \qquad \frac{d}{dx}G(s, x) = -\int_1^\infty \exp(-xy)y^{s+1}\frac{dy}{y} = -G(s+1, x).$$

Integration by parts, on the other hand, gives

$$(4) \qquad\qquad G(s+1, x) = \frac{\exp(-x)}{x} + \frac{s}{x}G(s, x).$$

Equations (3) and (4) give a nice recursive relation for all the derivatives $G^{(k)}(s, x)$ in terms of $G(s, x)$.

We will need bounds for $G(s, x)$ and its derivatives. For s in the critical strip, that is $0 < \mathrm{Re}(s) < 1$, we have

$$|G(s, x)| \leq \int_1^\infty \exp(-xy)y^{\mathrm{Re}(s)-1}dy.$$

Change the variables by $u = y - 1$ to get

$$|G(s, x)| \leq \exp(-x) \int_0^\infty \exp(-xu)(u + 1)^{\mathrm{Re}(s)-1}du.$$

Now $u + 1 \geq 1$ and by hypothesis, $\mathrm{Re}(s) - 1 < 0$, so $(u+1)^{\mathrm{Re}(s)-1} \leq 1$, and therefore

$$(5) \qquad |G(s,x)| \leq \exp(-x) \int_0^\infty \exp(-xu)du = \frac{\exp(-x)}{x}.$$

To estimate the derivatives $G^{(k)}(s,x) = (-1)^k G(s+k,x)$ this method does not apply, since we only have $\mathrm{Re}(s) + k - 1 < k$. Instead we can use the Cauchy formula for derivatives

$$\frac{|G^{(k)}(s,x)|}{k!} \leq \frac{M_R}{R^k},$$

where M_R is a bound for $G(s,w)$ with $|w - x| = R$. An estimate similar to (5) shows that $|G(s,w)| \leq \exp(-\mathrm{Re}(w))/\mathrm{Re}(w)$, which a calculus argument shows is maximized at $w = x - R$, that is,

$$(6) \qquad \frac{|G^{(k)}(s,x)|}{k!} \leq \frac{\exp(R-x)}{(x-R)R^k}$$

for any $0 < R < x$.

3. TRUNCATION OF THE *L*-SERIES.

To compute to D digits of accuracy, we need enough terms N in the *L*-series so that

$$(7) \qquad 2 \sum_{n>N} r_\phi(n) |\mathrm{Re}(G(s, 2\pi n/q^{1/2}))| \leq 10^{-D}.$$

Since the $r_\phi(n)$ are the coefficients of a weight one cusp form, $r_\phi(n) \ll n^{1/2}$. Using (5), the left side of (7) is

$$\ll \sum_{n>N} (q/n)^{1/2} \exp(-2\pi n/q^{1/2})$$

$$\approx q^{1/2} \int_N^\infty y^{-1/2} \exp(-2\pi y/q^{1/2}) dy$$

$$< (q/N)^{1/2} \int_N^\infty \exp(-2\pi y/q^{1/2}) dy$$

$$\ll q/N^{1/2} \exp(-2\pi N/q^{1/2}).$$

For this to be less than 10^{-D} we need

$$\log(q) + D\log(10) < 2\pi N/q^{1/2} + \log(N)/2;$$

it suffices that

$$(8) \qquad N = q^{1/2} \log(q \cdot 10^D)/2\pi.$$

4. ARRANGEMENT OF THE TAYLOR EXPANSIONS.

We seek to find intervals of radius Δ_j centered at x_j and bounding circles of radius R_j so that

$$0 < \Delta_j < R_j < x_j.$$

The tail of the Taylor expansion

$$\sum_{k=B}^{\infty} G_{j,k}(s)(2\pi n/q^{1/2} - x_j)^k$$

is bounded, via (6) by

$$(9) \qquad \sum_{k=B}^{\infty} \frac{\exp(R_j - x_j)}{x_j - R_j}(\Delta_j/R_j)^k = \frac{\exp(R_j - x_j)}{x_j - R_j} \frac{(\Delta_j/R_j)^B}{1 - \Delta_j/R_j},$$

where $G_{j,k}(s)$ is as in (1).

One might wish to minimize the total number of terms in all the Taylor expansions, but this seems intractable. Instead we will simply choose the truncation parameter B to be independent of j. To make this happen we require that

$$\begin{aligned} \Delta_j/R_j &= \quad \text{(constant)} \quad 1/c < 1, \\ x_j - R_j &= \quad \text{(constant)} \quad K > 0. \end{aligned}$$

Furthermore, in order that the endpoints of the intervals meet, we require that

$$x_j + \Delta_j = x_{j+1} - \Delta_{j+1}.$$

This gives

$$\begin{aligned} x_j &= R_j + K = c\Delta_j + K \\ x_{j+1} &= R_{j+1} + K = c\Delta_{j+1} + K, \end{aligned}$$

so

$$\begin{aligned} x_{j+1} - x_j &= c(\Delta_{j+1} - \Delta_j) \\ x_{j+1} - x_j &= \Delta_j + \Delta_{j+1}. \end{aligned}$$

Thus

$$\begin{aligned} \Delta_{j+1} + \Delta_j &= c(\Delta_{j+1} - \Delta_j) \\ \Delta_{j+1} &= \Delta_j(c+1)/(c-1). \end{aligned}$$

Now

$$x_{j+1} = x_j + \Delta_j + \Delta_{j+1} = x_j + \frac{2c}{c-1}\Delta_j$$

and

$$\Delta_j = R_j/c = (x_j - K)/c$$

implies that

$$x_{j+1} = x_j(c+1)/(c-1) - 2K/(c-1).$$

For simplicity we choose the parameters $c = 2$ and $K = x_1/2$ which gives

(10) $$x_j = \frac{x_1}{2}\left(3^{j-1} + 1\right),$$

and

(11) $$\Delta_j = 3^{j-1}\frac{x_1}{4}.$$

The simplest choice for x_1 is $2\pi/q^{1/2}$. This gives

$$I_j = \frac{2\pi}{q^{1/2}} \cdot \left[\frac{3^{j-1} + 2}{4}, \frac{3^j + 2}{4}\right].$$

Observe that I_1 contains only the $n = 1$ term, I_2 contains only the $n = 2$ term, I_3 contains only $n = 3, \ldots, 7$, etc. Thus for the very small values of j where the intervals contain fewer than B terms one can do better computing each term instead of using the Taylor expansion. This would improve the efficiency about 40% for q near 10^6, 30% near 10^9, and 20% near 10^{20}.

We can now determine how many intervals T are needed, since we need to compute terms in the series out to

$$n = N = q^{1/2}\log(q \cdot 10^D)/2\pi,$$

so

$$\frac{2\pi N}{q^{1/2}} = \log(q \cdot 10^D).$$

This requires that

$$x_T + \Delta_T = \left(\frac{3^T}{2} + 1\right)\frac{x_1}{2} > \log(q \cdot 10^D).$$

With $x_1 = 2\pi/q^{1/2}$ we want

$$2q^{1/2}\log(q \cdot 10^D)/\pi < 3^T,$$

or

$$T > \left\{\log(2/\pi) + \log(q^{1/2} \cdot \log(q \cdot 10^D))\right\}/\log(3).$$

With

$$\log(2/\pi) < 0, \quad \text{and} \quad 1/\log(3) < 1$$

we can simply choose

(12) $$T = \log(q^{1/2} \cdot \log(q \cdot 10^D))$$

5. TRUNCATION OF THE TAYLOR EXPANSIONS.

Suppose we make error δ in computing

$$G(s, 2\pi n/q^{1/2}) + G(1 - s, 2\pi n/q^{1/2}).$$

Using the bound $n^{1/2}$ for $r_\phi(n)$ and the rough estimate $q^{1/2}$ for N, the maximal error we make in the sum for $Z(t, \phi)$ is bounded by

$$q^{-1/4} \sum_{n=1}^{q^{1/2}} n^{1/2} \cdot \delta \ll \delta \cdot q^{1/2}.$$

Alternatively we can consider the standard error, assuming the errors in the terms are independent with standard deviation ϵ. Then the standard error in the sum is bounded by [2]

$$q^{-1/4} \left(\sum_{n=1}^{q^{1/2}} (n^{1/2} \cdot \epsilon)^2 \right)^{1/2} \ll \epsilon \cdot q^{1/4}.$$

We will assume the latter from now on. We want $\epsilon \cdot q^{1/4} < 10^{-D}$, or

(13) $$\epsilon < q^{-1/4} 10^{-D},$$

which will determine how many terms B we need to take in each Taylor expansion.

It follows from the choice made in §3 that

$$\frac{\Delta_j}{R_j} = \frac{1}{2} \qquad \text{and} \qquad x_j - R_j = x_1/2.$$

Thus the tail (9) of the Taylor series reduces to

(14) $$\epsilon = \frac{2 \exp(-x_1/2)}{x_1} 2^{1-B}.$$

Again, with $x_1 = 2\pi/q^{1/2}$ we combine (13) and (14) to find that we require

$$q^{1/2} \exp(-\pi/q^{1/2}) 2^{1-B}/\pi < q^{-1/4} 10^{-D}.$$

Since

$$\exp(-\pi/q^{1/2}) 2/\pi < 1$$

we ignore it, and so we need $q^{3/4} 10^D < 2^B$, and let

(15) $$B = 1.5 \cdot \log(q^{3/4} 10^D) > \log_2(q^{3/4} 10^D).$$

Even under the assumption of maximal error we would still only need $B = O(\log(q10^D))$.

6. IMPLEMENTATION AND EXAMPLES.

To implement this algorithm requires the computation of reduced representatives of all the forms, as well as their coordinates in terms of the generators of the cyclic factors of the class group. This computation is clearly $O(q^{1/2})$.

We then need to evaluate all the forms $ax^2 + bxy + cy^2$ on a rectangular grid of integer lattice points (x, y), large enough not to miss any representation of any integer $n < N$. Determining the dimensions of the grid is a Lagrange multipliers problem; we must maximize $g(x, y) = x$ (respectively $f(x, y) = y$) subject to the constraint $ax^2 + bxy + cy^2 = N$. One finds

$$x \leq 2(Nc/q)^{1/2} \qquad \text{and} \qquad y \leq 2(Na/q)^{1/2}.$$

Since the forms are reduced, the inequalities on c and a and our choice (8) of N imply

$$x \leq q^{1/4} \log(q \cdot 10^D)^{1/2} \qquad \text{and} \qquad y \leq (2/3^{1/4}) \log(q \cdot 10^D)^{1/2}.$$

We have triple (x^2, xy, y^2) for each lattice point, which are the rows of a $O(q^{1/4} \log(q \cdot 10^D)) \times 3$ matrix. Evaluation of the all the forms at all the lattice points consists of multiplying the above matrix by the $3 \times O(q^{1/2})$ matrix of form data. This is $O(q^{3/4} \log(q))$ multiplications (in terms of q, for a fixed number D of digits of accuracy.)

The representation numbers $r_Q(n)$ are computed by brute force and ignorance; we look at each entry in the matrix product and increment the corresponding $r_Q(n)$, another $O(q^{3/4} \log(q))$ operations.

Fast Fourier Transform on a group of size $O(q^{1/2})$ is $O(q^{1/2} \log(q^{1/2}))$, and the function is vector valued with $N = O(q^{1/2} \log(q))$ entries, for a total computation of size $O(q \log(q)^2)$.

For each character ϕ, the precomputation of the sums

$$\sum_{n \in I_j} r_\phi(n) \frac{(2\pi n/q^{1/2} - x_j)^k}{\Delta_j^k}$$

takes $N \cdot B$ operations, which is $O(q^{1/2} \log(q)^2)$ by (8) and (15). Subsequently, evaluations of $Z(t, \phi)$ using (2) require only $T \cdot B$ operations, which is $O(\log(q)^2)$ by (12) and (15).

This algorithm was implemented in *Mathematica* and run on a 400 MHz. Apple Powerbook G4 under OS X 10.2[1]. *Mathematica* is a good

---

[1]This paper was written in June 2003.

-q	h(-q)	C(-q)	♯ L-functions
-10000003	706	{706}	352
-10000004	1648	{412, 2, 2}	820
-10000007	3660	{3660}	1829
-10000011	816	{204, 2, 2}	404
-10000015	1134	{1134}	566
-10000019	1275	{1275}	637
-10000020	928	{232, 2, 2}	460
-10000023	2064	{258, 2, 2, 2}	1024
-10000024	990	{330, 3}	494
-10000027	282	{282}	140
-10000031	5426	{5426}	NA
-10000036	876	{876}	437
-10000039	1912	{956, 2}	954
-10000043	618	{618}	308
-10000047	1512	{756, 2}	754
-10000051	742	{742}	370
-10000052	1692	{846, 2}	844
-10000055	3584	{896, 2, 2}	1788
-10000056	1480	{370, 2, 2}	736
-10000059	968	{484, 2}	482
-10000063	1722	{1722}	860
-10000072	724	{724}	361
-10000079	4147	{4147}	2073
-10000083	416	{208, 2}	206
-10000084	1364	{682, 2}	680
-10000087	1076	{1076}	537
-10000088	1512	{126, 6, 2}	752
-10000091	1382	{1382}	690
-10000095	2928	{732, 2, 2}	1460
-10000099	640	{320, 2}	318

TABLE 1. Class group data for various discriminants

choice if one wants to re-invent the wheel as few times as possible. The incomplete Gamma function $\Gamma(s, x)$ is available; it is computed via hypergeometric functions and continued fractions according to [9, A.9.4]. Also the Fast Fourier Transform is supported in a sophisticated way [9, A.9.4], via decomposition of the length of lists of data into prime factors. For large factors, fast convolution methods are
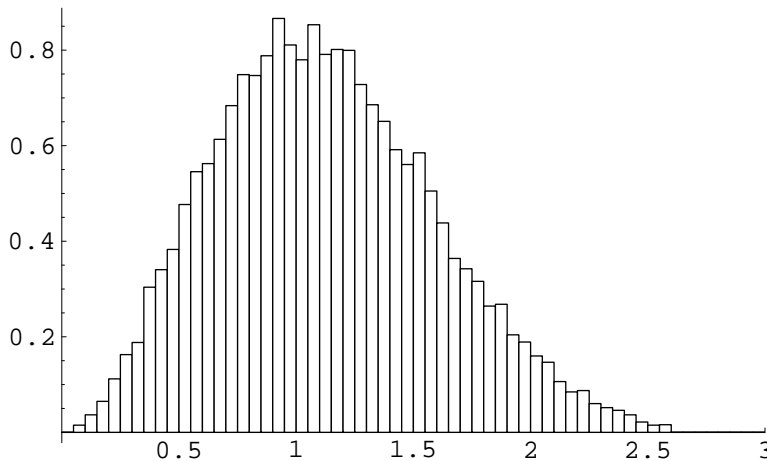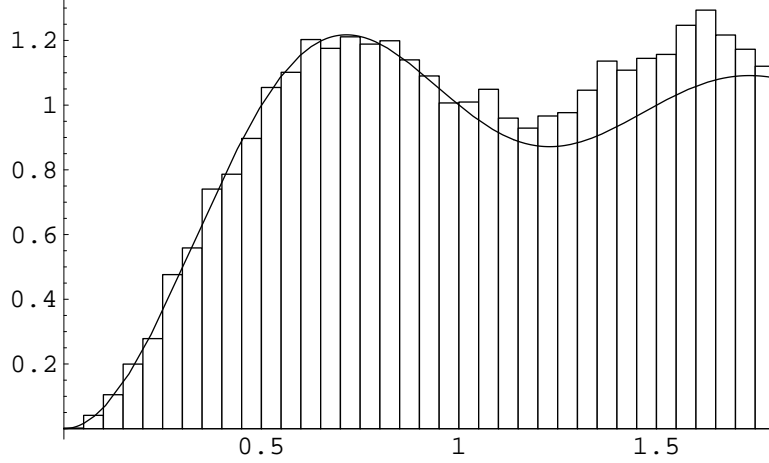
FIGURE 1. Histogram data of first zero above 0.

used. Since our data are real (the representation numbers $r_Q(n)$), the algorithm makes use of a real transform method. Non-cyclic abelian groups are handled automatically. The *Mathematica* function `FindRoot` finds zeros of functions by a combination of damped Newton's method, the secant method, and Brent's method [9, A.9.4].

On the other hand, a specialized package for number theory, such as PARI [13] is preferable for computations involving binary quadratic forms. Fortunately, *Mathematica*'s MathLink capability allows users to install their own C code, and this was done, essentially installing all the PARI functions for computations with binary quadratic forms into *Mathematica*. (This was documented in [14].)

The accuracy of the algorithm was checked by computing, to 15 digits, the zeros of some genus character *L*-functions. By genus theory, these are products of Dirichlet *L*-functions attached to quadratic characters. These zeros were first computed directly, writing the Dirichlet *L*-function as a linear combination of Hurwitz zeta functions (also supported in *Mathematica*). The zeros were also compared to the data in [16], and agreed to the number of digits given.

The algorithm was then run on 21336 *L*-functions with conductor near $q = 10^7$. The Hardy function $Z(t, \phi)$ was evaluated, to six digits of accuracy, for t between $0.$ and $1.$ in steps of size $2\pi/(20 \log(q))$, looking for sign changes. When detected, *Mathematica*'s `FindRoot` was used to find the zero. This calculation took 137 hours, and found 35190 zeros below $t = 1$. No attempt was made to prove GRH for

FIGURE 2. 1-level density vs. $1 - \sin(2\pi x)/(2\pi x)$

these L-functions in this range, or to prove that all the zeros had been located. The zeros themselves are available at

`http://www.math.ucsb.edu/~stopple/`

Table 1 shows the discriminants, class numbers, class group structure, and number of L-functions (omitting real (genus) characters, and one of each complex conjugate pair.) Figure 1 shows a histogram of of the lowest zero, renormalized by $\tilde{\gamma} = \gamma \cdot \log(q)/(2\pi)$. These seem to support a conjecture that the corresponding random matrix model is symplectic (see [7, Figure 5]). The mean height above $0.$ is 1.13, considerably larger than

$$0.78 = \int_0^\infty t\nu_1(USp)(t)dt$$

predicted by this model, due to the slow convergence as $q \to \infty$. Figure 2 compares the 1-level density of $USp(\infty)$, that is,
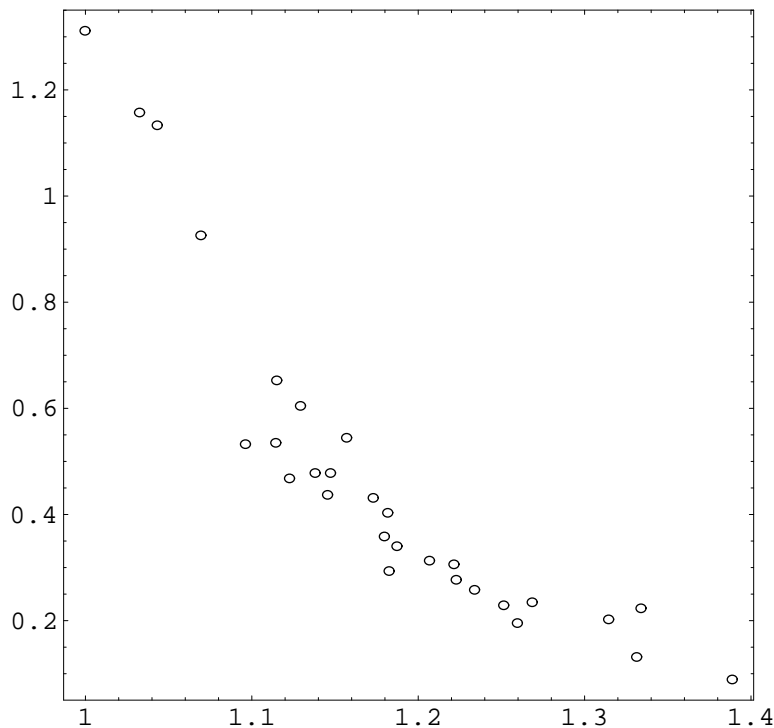
$$1 - \sin(2\pi x)/(2\pi x),$$

to the histogram data

$$\frac{(\beta - \alpha)^{-1}}{\sharp\{\text{characters } \phi\}} \sum_\phi \sum_\gamma \chi_{[\alpha,\beta)}(0.78/1.18 \cdot \tilde{\gamma}),$$

where

$$[\alpha, \beta) = [0, .05), [.05, 0.1), \ldots, [1.75, 1.80)$$

FIGURE 3. Mean of first zero vs. $h(-q)/\sqrt{q}$.

The zeros $\tilde{\gamma}$ have been re-renormalized so that the mean above $0.$ of the first zero is 0.78.

Finally, for each of the 29 discriminants considered, the mean above $0.$ of the first zero for the *L*-functions with that discriminant was compared to the relative size $h(-q)/\sqrt{q}$ of the class number. Figure 3 shows the plot. The discriminants with smaller class number tend to have a *larger* mean first zero. The correlation between the two quantities is -0.89. The connection between this phenomenon and the Chowla-Selberg formula is discussed in §3 of [1].

As mentioned in the introduction, this method can be adapted to compute other types of *L*-functions. For an elliptic curve *L*-function, one gives up the advantage of the Fast Fourier Transform, but gains the advantage of doing the precomputations with integer arithmetic. The elliptic curve

$$E : x^3 + y^3 = 763002$$

has rank 5 according to [3], so by the Birch-Swinnerton Dyer conjectures the *L* functions $L(s, E)$ should have a zero of order 5 at $s = 1$.
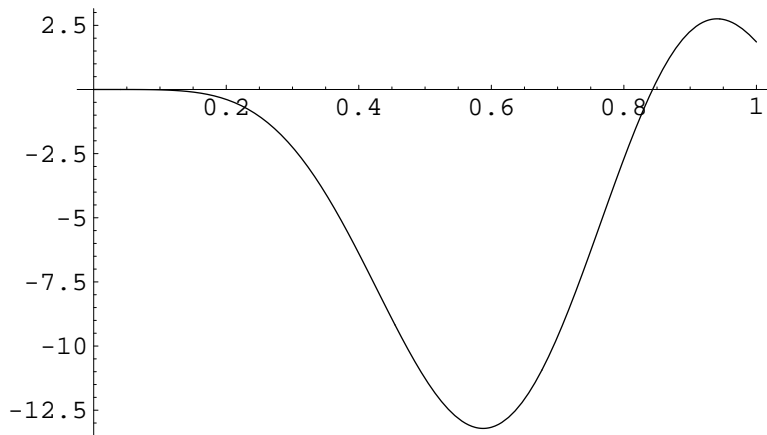
FIGURE 4. Elliptic curve L-function with conjectural order 5 zero at $s = 1$ (i.e. $t = 0$).

The conductor is 1746516156012. Figure 4 show a plot of the corresponding Hardy function $Z(t, E)$.

Low lying zeros of L-functions for quadratic Dirichlet characters χ_d are useful and scarce. The algorithm described here is of no real use since the existence of such a zero can be ruled out by a single evaluation. However, highly accurate values of zeros have been useful in the past, for example [11]. With the parameter $D = 110$, and discriminant $d = -175990483$ we can compute $L(1/2 + it, \chi_d)$ is zero,

$$t = 0.00047524399542016290008767557526752$$
$$446841851348862432434240449427326484$$
$$628127211844705565445126704808839630.$$

This was checked with the method of [17], and is accurate to more than 100 digits.

REFERENCES

[1] C. Bays, K. Ford, R. Hudson, and M. Rubinstein, *Zeros of Dirichlet L-functions near the real axis and Chebyshev's bias*, J. Number Theory, **87**, 2001, pp. 54-76.

[2] G. Dahlquist, Numerical Methods, Prentice Hall, 1974.

[3] N. Elkies and N. Rogers, *Elliptic curves $x^3 + y^3 = k$ of high rank*, in Algorithmic Number Theory, Lecture Notes in Comput. Sci., **3076**, Springer, Berlin, 2004, pp. 184-193.

[4] E. Fouvry and H. Iwaniec, *Low-lying zeros of dihedral L-functions*, Duke J. Math., **116**, 2003, pp. 189-217.

[5] H. Iwaniec, Topics in Classical Automorphic Forms, Graduate Studies in Mathematics **17**, AMS, 1991.

[6] H. Iwaniec and E. Kowalski, Analytic Number Theory. AMS Colloquium Publications **53**, 2004.

[7] N. Katz, and P. Sarnak, *Zeros of zeta functions and symmetry*, Bull. Amer. Math. Soc., **36**, no. 1, 1999, pp. 1-26.

[8] J. Lagarias, and A. Odlyzko, *On computing Artin L-functions in the critical strip*, Math. Comp. **33**, 147, 1979, pp.1081-1095.

[9] S. Wolfram, The Mathematica Book, 4th ed., 1999.

[10] H. Montgomery, and P. Weinberger, *Notes on small class numbers*, Acta Arith. **XXIV** 1974, pp. 529-542.

[11] A. Odlyzko and H. te Riele, *Disproof of the Mertens' conjecture*, J. Reine Angew. Math. **357** (1985), pp. 138-160.

[12] A. Odlyzko and A. Schönhage, *Fast algorithms for multiple evaluations of the Riemann zeta function*, Trans. Amer. Math. Soc. **309**, 2, 1988 pp. 797-809.

[13] PARI-GP. By C. Batut, D. Bernardi, H. Cohen , and M. Olivier, currently maintained by K. Belabas. `http://www.parigp-home.de`

[14] PARI-GP mailing list, available at `http://www.parigp-home.de/lists/200212aav`

[15] M. Rubinstein, *Evidence for a spectral interpretation of the zeros of L-functions*, Ph.D. thesis, Princeton 1998.

[16] R. Rumely, *Numerical computations concerning the ERH*, Math. Comp., **61**, 203, 1993, pp. 415-440.

[17] P. Weinberger, *On small zeros of Dirichlet L-functions*, Math. Comp., **29** no. 129, 1975, pp. 319-328.

E-mail address: `stopple@math.ucsb.edu`